

Funktion	Code			Beschreibung		
	Hex	Char.	Parameter	CMD + Daten (->RS232)	Antwort (<- RS232)	
GetSlaveAdress*	0x61	a		'a'+<CR>	'61'+ 'XX' +<CR>	Slave Adresse anfordern
ChangeSlaveAdress *	0x63	c	d ₁	'c'+ 'XX' +<CR>	'63'+<CR>	Slave Adresse ändern
SaveSlaveAddress	0x73	s		's'+<CR>	'73'+<CR>	Slave Adresse speichern
WriteData **	0x77	w	d ₁ ,d ₂ ,...d ₁₂₈	'w'+ 'XXYYZZ....' +<CR>	'77'+<CR>	Daten ins RAM schreiben ohne Interrupt
WriteDataInt1	0x78	x	d ₁ ,d ₂ ,...d ₁₂₈	'x'+ 'XXYYZZ....' +<CR>	'78'+<CR>	Daten ins RAM schreiben mit Interrupt 1
WriteDataInt2	0x79	y	d ₁ ,d ₂ ,...d ₁₂₈	'y'+ 'XXYYZZ....' +<CR>	'79'+<CR>	Daten ins RAM schreiben mit Interrupt 2

Sendet ein Master dem I2C-RS232-Adapter (Slave) Daten, so werden diese zwischengespeichert, konvertiert und anschließend über die serielle Schnittstelle ausgegeben. Die Daten werden wie folgend ausgegeben:

I2CdataAvail	0x69	i	d ₁ ,d ₂ ,...d ₁₂₈		'69'+ 'XXYYZZ....' +<CR>	I2C-Daten empfangen
--------------	------	---	---	--	---------------------------------	---------------------

Alle gesendeten bzw. empfangenen Daten werden immer mit einem CarriageReturn (<CR> = 0x0D HEX) abgeschlossen.

*) ChangeSlaveAdress / GetSlaveAdress:

XX ist die Slave Adresse. Das Byte wird als Zeichenkette in Hexdezimaler Form übergeben.

z.B. die Adresse eines PCF8574 ist 0x40. Diese wird wie folgend übergeben: '**c40**' + <CR>

***) WriteData:

XXYYZZ... sind die zu übertragenden Daten. Es dürfen mindesten 1 Byte und maximal 128 Bytes übertragen werden.

Ein Beispiel: es sollen die 5 Bytes 0xA1, 0x1F, 0x22, 0x5C, 0xB0 zum Adapter gesendet werden, dann wird folgende Zeichenkette abgeschlossen mit einem CarriageReturn über die Serielle Schnittstelle gesendet: '**wA11F225CB0**' + <CR>

RS232 – Einstellungen:

Baud: 19200
 DataBits: 8
 StopBits: 1
 Parity: None